CIIPS Glory
Small Soccer Robots with Local Image Processing

Thomas Bräunl, Petter Reinholdtsen, Stephen Humble
Centre of Intelligent Information Processing Systems (CIIPS)
Dept. of Electrical and Electronic Engineering
The University of Western Australia, Nedlands, Perth, WA 6907

Abstract. CIIPS Glory is using a local intelligence approach to solve the task of robot soccer. In this respect our system is much closer to the F-2000 league (mid-size) than other approaches in the F180 (small-size) league. We are using *EyeBot* controllers, the *RoBIOS* operating system and the *EyeCam* digital on-board camera systems (CMOS). Our robot family also comprises 6-legged and biped walking machines, an omni-directional robot and an autonomous air-plane.

1 Introduction

The CIIPS Glory robot soccer team (Figure 1) has competed in RoboCup tournaments in Singapore (1998, [1]), Melbourne (1999), and Melbourne (2000). In the first two events it was the only team using a local intelligence approach without employing global vision, while at Melbourne 2000 three teams used this approach. Since the basic robot design has already been published in [2], we will concentrate in this article on innovations and improvements of our design.

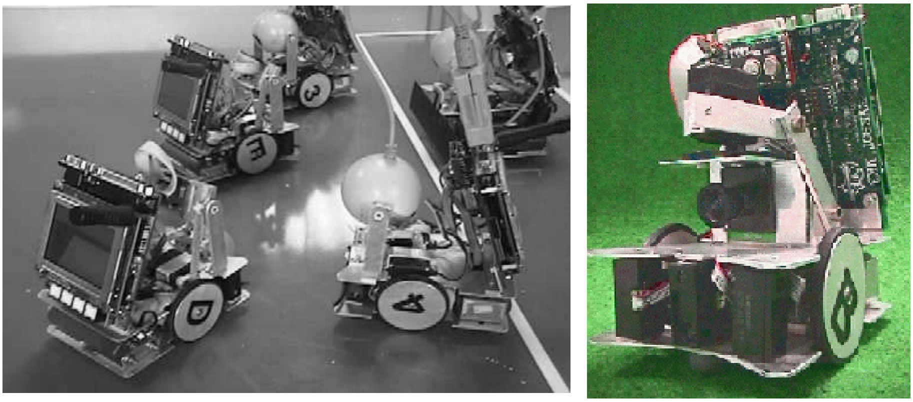

Fig 1. *EyeBot* soccer robots with local intelligence

We experienced major difficulties in the past robot soccer competitions, which lead to a number of extensions and modifications of our existing robot design.

Problems	Solutions
• Frame-rate too slow	• Integrate FIFO buffer in camera interface
• Poor self-localization	• Integrate compass sensor
• Poor ball-tracking behaviour	• Change from tilting to panning camera
• Limited robot interaction	• Add optional wireless network

P. Stone, T. Balch, and G. Kraetzschmar (Eds.): RoboCup 2000, LNAI 2019, pp. 523-526, 2001.

2 Team Members

A number of people contributed to the CIIPS Glory 2000 team, of which Petter Reinholdtsen, Mari Wang, and Stephen Humble went to the competition in Melbourne. Petter Reinholdtsen, team leader and volunteer from Norway, spent a year at UWA to develop the on-board software. Mari Wang, student at UWA, developed the goalkeeper software. Stephen Humble, volunteer from Australia, worked on both mechanics and software. Philippe Leclercq, Ph.D. student at UWA, developed the image processing library. Mark Gaynor, volunteer from Australia, helped with the radio communication and electronics. Klaus Schmitt from Univ. Kaiserslautern developed operating system routines and sensor drivers. Thomas Bräunl worked on operating system routines and is director of the Mobile Robot Lab at UWA.

3 Hardware Platform

Each of our robots comprises a microcontroller system (*EyeBot*) interfaced to a digital color camera, distance sensors, shaft encoders, compass, DC motors, servos, wireless module, and a graphics display. All image processing is done on-board. We are interested in research on autonomous mobile systems, so we took this clearly disadvantaged robot soccer approach instead of using a global overhead camera.

Since we rely on local image processing on a microcontroller, we had to make concession about image resolution (80x60 pixels in 32 bit color). On the other hand, we could not afford a too low frame rate, or we would see a passing ball only once in a frame sequence and could not tell which direction it went. Directly interfacing a digital CMOS camera to the CPU allows to do image processing at a rate of 3.5 frames per second (fps). Only after the competition we finished integrating a hardware buffer, which significantly reduces the number of time consuming interrupts required for data acquisition. This now gives us a framerate of about 15 fps.

Self-localization is an important task for our robots, since we do not use a global positioning system like an overhead camera. We rely on dead reckoning from a specified starting position and orientation. However, a robot will soon lose its exact position and orientation due to wheel slippage or - much worse - collision with another robot. Therefore, we integrated a digital compass to our robots. In the robot soccer application orientation is more important than position, because it guarantees that a robot is heading for the right goal. The robot position can be updated by its local infrared sensors whenever it gets close to one of the side walls or a corner.

The camera mechanics was changed from tilting to panning in order to improve ball tracking. The camera can be moved sideways at a higher speed than the whole robot can turn, so this will allow us tracking of balls moving faster across a robot's field of view than it would be possible with a static camera mount.

An implementation-independent soccer protocol allows us to test different implementations of the robot control programs side-by-side.

4 Wireless Communication

There are several reasons for a communication network in a multi-robot scenario:

a. To allow robots to communicate with each other
b. To remote-control one or several robots
c. To monitor robot sensor data
d. To run a robot with off-line processing (combine previous points)
e. To create a multi-robot monitoring console

Our approach accomplishes all these requirements by using a "virtual token ring" structure [6]. At the initialization of the network, one robot takes on the task as "ring master" and determines which other robots want to participate in the communication system. Each robot (and each base station) is identified by a unique ID number, which is being used for addressing a robot. A specific ID number is used for broadcasts, which follows exactly the same communication pattern. In the same way, subgroups of nodes can be implemented, so a group of robots receives a particular message.

The "virtual token ring" system enables a number of robots to exchange data with one another over the same frequency by taking turns in data transmission - only the robot which holds the token is allowed to transmit. Transmitted data includes ball detection data (current ball position and velocity vector), local robot position and orientation, and plan details to be shared with other robots.

5 Self-Localization

The newly developed self-localization method combines short range distance sensor information with dead reckoning from the wheel encoders. It is based on ideas from [3], [4] and [5] with additional consideration for range sensors with limited accuracy and reach. The robot collects distance information from the sensors synchronized with the camera frame rate at 3.7 times per second. The sensor information is stored in a list with a time stamp and the position given by read reckoning. This list is then processed to find lines which are matched against the soccer field map to estimate the current position (see Figure 2).

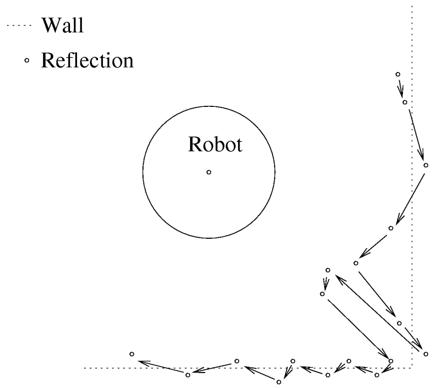

Fig 2. Robot self-localization

Processing is done as follows: First, points are sorted based on their angle from a well chosen origin. The origin should be inside the soccer field and have no obstacles (or as few as possible) between itself and the walls, e.g. the current robot position. The angles between the points are then used in sorted order to calculate an angle histogram. This groups points into histogram classes. Points are then sorted depending on the number of members in the histogram class and the angle from the origin. This way, points located on the same wall should be grouped together. The points are then traced to find lines, starting with the first point in a group. The next point is added to the line if the angle between the two points is in the same histogram group as the current point. The end result is a list of lines which are matched with the map of the soccer field. To avoid lines which are not part of a wall, line segments shorter than 20 cm are ignored.

This method works quite well, but requires some time to process the list. A robot blocks for about 10 seconds every time the list of points is processed. Clearly, this makes this self-localization method of limited value in a real-time competition environment. Further problems arose from uncalibrated sensors, inaccurate sensor readings, sensor reflections and incorrect readings from angled walls.

Acknowledgments

The authors would like to acknowledge the work of all students who participated in the previous CIIPS robot soccer campaigns: Birgit Graf for the first soccer team 1998, who also proof-read this paper, Andrew Barry and Ilario Dichiera for the soccer team 1999. More information is available on the Internet, where also the source code of the soccer program and a robot simulator are available:

`http://www.ee.uwa.edu.au/~braunl/eyebot/`

References

1. Th. Bräunl, B. Graf, *Robot Soccer with Local Vision*, Pacific Rim International Conference on Artificial Intelligence, Singapore, Nov. 1998, pp. 14–23 (10)
2. Th. Bräunl, B. Graf, *Small Robot Agents with On-Board Vision and Local Intelligence*, Advanced Robotics, vol. 14, no. 1, 2000, pp. 51–64 (14)
3. K. Graves, W. Adams, and A. Schultz, *Continuous localization in changing environments*, in Proceedings of the IEEE Int. Symposium on Computational Intelligence in Robotics and Automation, pages 28--3, 1997.
4. J.-S. Gutmann, T. Weigel and B. Nebel, *Fast, Accurate, and Robust Self-Localization in Polygonal Environments*, in: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '99), Kyongju, Korea, 1999
5. A. Schultz, W. Adams, and B. Yamauchi, *Integrating Exploration, Localization, Navigation and Planning With a Common Representation*, Autonomous Robots, 6 (3), pp. 293-308, 1999
6. P. Wilke, Th. Bräunl, *Flexible Wireless Communication Network for Mobile Robot Agents*, Industrial Robot International Journal, to appear, 2000